What is claimed is:

1. A method for filtering data, the method comprising:

receiving a plurality of data samples;

computing a locus of the samples;

normalizing a value of an input sample to a range centered on the locus;

passing the data through a distance-based filter ; and

normalizing an output value of the distance-based filter to a predetermined output range.

2. The method of claim 1 wherein the distance-based filter further comprises a median filter.

3. The method of claim 1 wherein the distance-based filter further comprises a low-pass filter.

4. The method of claim 1 wherein the distance-based filter further comprises one of a band-pass filter and a high-pass filter.

5. The method of claim 1 wherein computing a locus of the samples comprises computing an average of a last two of the samples.

6. The method of claim 1 wherein computing a locus of the samples comprises computing one of an arithmetic mean, a geometric mean, a harmonic mean, and a quadratic mean of the samples.

7. The method of claim 1 wherein computing a locus of the samples comprises computing an average of a last two of the samples together with the input sample.

8. The method of claim 1 wherein computing a locus of the samples comprises computing an average of a last three of the samples.

9. The method of claim 1 wherein computing a locus of the samples comprises computing an average of a last three of the samples together with the input sample.

10. The method of claim 1 wherein computing a locus of the samples comprises selecting a previous filter output value.

11. A method for filtering data , the method comprising:

determining a current locus of a plurality of data samples as a function of signal history;

determining a current normalizing range as a function of the current locus;

normalizing an input value to the current normalizing range;

passing the input value and the current locus through a distance-based filter; and

normalizing an output value of the distance-based filter to the current normalizing range.

12. The method of claim 11 wherein the distance-based filter is one of median filter, a low-pass filter, a high-pass filter, and a band-pass filter.

13. The method of claim 11 wherein determining a current locus of the data samples further comprises determining one of an arithmetic mean, a geometric mean, a harmonic mean, and a quadratic mean of the data samples.

14. A filter device, comprising:

a means for receiving a plurality of data samples;

a means for computing a locus of the samples;

a means for computing a distance between an input value and the locus; and

a means for determining an output value as a function of a difference between the input value and one of the plurality of data samples.

15. The filter device of claim 14, further comprising a means for comparing the distance between the input value and the locus with a predetermined threshold value.

16. The filter device of claim 15, further comprising a means for normalizing the distance between the input value and the locus when the distance exceeds a predetermined limit.

17. The filter device of claim 16 wherein normalizing the distance between the input value and the locus includes adjusting the sample to be within one-half circle of the locus.

18. The filter device of claim 14, further comprising a means for comparing the output value with a predetermined threshold value.

19. The filter device of claim 18, further comprising a means for normalizing the output value when the output value exceeds a predetermined limit.

20. The filter device of claim 19 wherein normalizing the output value includes adjusting the output value to be within a predetermined output range.

21. A system for circular distance normalization of filtered data, the system comprising:

a) a first memory for storing a plurality of machine instructions;

b) a second memory for storing a plurality of data samples; and

c) a processor coupled to the first and second memories, the processor executing the plurality of machine instructions to implement a plurality of functions, the functions including:

i) processing at least a portion of the plurality of data samples to compute a locus of the samples;

ii) computing a distance between an input value and the locus;

iii) determining an output value by computing a difference between the input value and one of the plurality of data samples; and

iv) providing the output value.

22. The system of claim 21 wherein the function of determining an output value includes applying a distance-based filter to the plurality of data samples.

23. The system of claim 21 wherein the functions executed by the processor further include normalizing the distance between the input value and the locus when the distance exceeds a predetermined threshold value.

24. The system of claim 21 wherein the functions executed by the processor further include normalizing the output value when the output value exceeds a predetermined threshold value.

25. The system of claim 21 wherein the function of processing at least a portion of the plurality of data samples to compute a locus of the samples includes computing an approximation of the locus of the samples.

26. The system of claim 25 wherein computing a locus of the samples comprises computing an average of data samples retrieved from the second memory.

27. The system of claim 21 wherein computing a locus of the samples comprises computing an average of a plurality of recent data samples.

28. A computer program product for filtering data, wherein the computer program product comprises:

a computer-readable storage medium; and

computer-readable program code means embodied in the medium, the computer-readable program code means comprising:

first computer-readable program code means for determining a locus of a received plurality of data samples,

second computer-readable program code means for normalizing a value of an input sample to a range centered on the locus determined from the first computer-readable program code means,

third computer-readable program code means are included for distance-based filtering of the data, and

fourth computer-readable program code means are included for normalizing an output value of the distance-based filter to a predetermined output range.

29. The computer program product of claim 28 wherein the first computer-readable program code means determines the locus of the samples by determining an average of at least a portion of the data samples.

30. The computer program product of claim 28 wherein the first computer-readable program code means determines the locus of the samples by determining an average of at least a last two of the data samples.

31. The computer program product of claim 28 wherein the first computer-readable program code means determines the locus of the samples by determining an average of at least a last two samples together with the input sample.

32. The computer program product of claim 28 wherein the first computer-readable program code means determines the locus of the samples by determining an average of at least a last three of the data samples.

33. The computer program product of claim 28 wherein the first computer-readable program code means determines the locus of the samples by determining an average of at least a last three of the data samples together with the input sample.

34. The computer program product of claim 28 wherein the first computer-readable program code means determines the locus of the samples by selecting a previous filter output value.

35. The computer program product of claim 28 wherein the distance-based filtering of the third computer-readable program code means further comprises a median filtering.

36. The computer program product of claim 28 wherein the first computer-readable program code means determines the locus of the samples by computing relative to at least a portion of the samples one of an arithmetic mean, a geometric mean, a harmonic mean, and a quadratic mean.

37. A method for filtering data , the method comprising:

buffering incoming sample data;

using an ordered compare operation to compare each buffered data sample to each other data sample one time;

aggregating results of the compare operations into a bit-array;

retrieving an index from a lookup table as a function of the bit-array; and

returning the buffer element from the input sample buffer selected by the index.

38. The method of claim 37 wherein the incoming sample data is stored using a circular buffer.

39. The method of claim 37 wherein the incoming sample data is stored using a shift register.

40. A system for filtering data, the system comprising:

a) a memory for storing a plurality of machine instructions;

b) a buffer for storing a plurality of data samples; and

c) a processor coupled to the memory and the buffer, the processor executing the plurality of machine instructions to implement a plurality of functions, the functions including:

i) buffering incoming sample data;

ii) using an ordered compare operation to compare each buffered data sample to each other data sample one time;

iii) aggregating results of the compare operations;

iv) storing results of the compare operations into a bit-array;

v) indexing into a lookup table as a function of the bit-array; and

vi) returning from the lookup table an index of the buffer.

41. The system of claim 40 wherein the buffer is structured as a conventional circular buffer.

42. The system of claim 41 wherein the incoming sample data is stored using a circular buffer.

43. A filter for filtering data , the filter comprising:

a buffer means for buffering incoming sample data;

an ordered comparing means for comparing each buffered data sample to each other data sample one time;

an aggregating means for aggregating results output by the ordered comparing means;

a means for storing the results output by the ordered comparing means into a bit-array;

a means for indexing into a lookup table as a function of the bit-array; and

a means for returning from the lookup table an index of the buffer element.

44. The filter of 43 wherein the buffer means is structured as a circular buffer.

45. The filter of claim 43 further comprising a storage means for storing the incoming sample data.

46. The filter of claim 45 wherein the storage means is structured as a circular buffer.

47. A computer program product for filtering data, wherein the computer program product comprises:

a computer-readable storage medium; and

computer-readable program code means embodied in the medium, the computer-readable program code means comprising:

first computer-readable program code means for buffering incoming sample data,

second computer-readable program code means for operating an ordered compare function to compare one time each buffered data sample to each other data sample,

third computer-readable program code means for aggregating results of the compare operations,

fourth computer-readable program code means for storing results of the compare operations into a bit-array,

fifth computer-readable program code means for indexing into a lookup table as a function of the bit-array, and

sixth computer-readable program code means for returning an index of the buffer from the lookup table.

48. The computer program product of claim 47 wherein the first computer-readable program code means utilizes a buffer that is structured as a circular buffer.